



链滴

PAT 甲级刷题实录——1006

作者: [aopstudio](#)

原文链接: <https://ld246.com/article/1578973967804>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原题链接

<https://pintia.cn/problem-sets/994805342720868352/problems/994805516654460928>

思路

这题思路很简单，先读第一个人的in和out时间，暂存为unlock和lock的时间，并把该人同时暂记为unlocker和locker，之后把每个人的in和out时间同unlock和lock的时间作比较，如果in的时间比unlock早，那么就把unlock的时间更新为in的时间，并把unlocker更新为这个人，如果out的时间比lock更，那么就把lock的时间更新为out的时间，并把locker更新为这个人。重点在于如何进行时间的比较。我一开始想着是把时、分、秒分别读入，并且分开来比较，结果写了一个完全错误的逻辑（后文会提）。结果匪夷所思的是，这个完全错误的逻辑在PAT的评测系统里竟然只有一个测试用例过不去，其他的竟然全部通过。也正因为这个原因，我一开始坚信自己的逻辑没有错误。后来上牛客网的评测系统测发现一个测试用例都过不去，这才觉得可能逻辑上确实有点问题。后来上网一搜，发现其实不用这麻烦比较。C++可以直接对string进行比较，大小是根据string的字典顺序，而时间的早晚实际上是合字典顺序的，因此可以用这个方法，既简单准确率又高。先写一下正确的代码吧：

代码

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    int total;
    char c;
    string unlocker, locker, nowPerson, unlockTime, lockTime, inTime, outTime;
    cin >> total;

    for (int i = 0; i < total; i++)
    {
        if (i == 0) //初始化
        {
            cin >> unlocker >> unlockTime >> lockTime;
            locker = unlocker;
        }
        else
        {
            cin >> nowPerson >> inTime >> outTime;
            if (inTime < unlockTime)
            {
                unlockTime = inTime;
                unlocker = nowPerson;
            }
            if (outTime > lockTime)
            {
                lockTime = outTime;
                locker = nowPerson;
            }
        }
    }
    cout << unlocker << ' ' << locker;
```

```
    return 0;
}
```

错误示范

一开始脑子完全短路，把时分秒分别判断并且用且连接起来，就像这样`inHour<=unlockHour&&inMinute<=unlockMinute&&inSecond<unlockSecond`，这个错误在于只有当in的三个时间数值都比unlock的小时才能更新，但实际上并不需要，这三个的优先级是递减的，即当inHour比unlockHour小时，就不需要考虑后面两个数值了，minute更小的时候也不需要考虑second了。

补充知识

虽然这题里面不需要分别输入hour,minute和second，但说不定以后的题目里面会需要对三个数值进行比较，因此我把分别输入三个数值的方法也记在这里。对于HH:MM:SS格式的数据，C语言和C+都没有自动读取三个数值的方法，只能手动处理输入。C语言的代码如下：

```
scanf("%d:%d:%d", &hour, &minute, &second);
```

C++有好几种方法，其中最简单的方法是定义一个char类型变量用来存放读入的:，C++处理输入的代码如下：

```
char c;
cin >> hour >> c >> minute >> c >> second;
```

应该说C语言的输入可读性更高，一看就知道什么意思，C++的cin方式需要另外定义一个变量，浪费空间的同时可读性也不高。由此推翻了我之前提出的cin完全优于scanf的观点，我会在《充分利用C++优势做算法题》这篇文章中做出修改。